Deep learning and natural language processing for innovation detection in FinTech

by

Mihai Dobri

M.S., University Politehnica of Bucharest, Romania, 2015

A THESIS

submitted in partial fulfillment of the
requirements for the degree

MASTER OF SCIENCE

Department of Computer Science
Carl R. Ice College of Engineering

KANSAS STATE UNIVERSITY
Manhattan, Kansas

2020

Approved by:

Major Professor
Dr. Doina Caragea

Copyright

Abstract

Advancements in technology have resulted in the emergence of numerous FinTech innovations. However, a global understanding of such innovations is limited, due to a lack of an underlying taxonomy and benchmark datasets in the FinTech domain. To address this limitation, we develop a FinTech taxonomy and manually annotate a set of FinTech patent abstracts according to the taxonomy. We use the annotated dataset to train deep learning models. Experimental results show that the deep learning models can accurately identify FinTech innovations. Specifically, we focus on patent document classification, and explores the predictive capabilities of three document sections alone and in combination. Our results indicate that the title and abstract in combination are most efficient in detecting FinTech innovations.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to acknowledge the support and expertise of my advisor, Dr. Doina Caragea, as well as assistance from Dr. Theodor Cojoianu, Kyle Glandt and George Mihaila. This thesis would not have possible without their guidance.

Chapter 1

Introduction

Patent documents represent a vast resource for research, litigation, and other uses, and yet searching through millions of documents can be time-consuming or prohibitively expensive. This is particularly true if one wishes to search for categories that are non-standard in patent offices. For instance, currently FinTech patents, which combine financial and technology domains, are not categorized as such by patent offices. This means trying to gain insight into FinTech innovation patterns is difficult. To address this, our research uses deep-learning models to attempt novel classification of patent documents into subcategories of FinTech using a taxonomy in accordance to FinTech literature.

The financial and technology sectors have been interwoven in the last 150 years [4], ever since the communication infrastructure underpinning financial transactions has been built. Post-2008, the year marking the all-time low trust in financial institutions, advancements in technology and data science have resulted in the emergence of numerous FinTech start-ups and placed start-ups, as well as highly trusted technology sector incumbents in a good position to challenge the traditional financial sector in the provision of financial services [5, 6].

According to KPMG [1], in 2019, global investments in FinTech start-ups attracted $135.7B with 2,693 deals, most notably in FinTech sub-sectors such as payment technologies and in-

[1] https://assets.kpmg/content/dam/kpmg/xx/pdf/2020/02/pulse-of-fintech-h2-2019.pdf

vestment and lending platforms. FinTech innovations do not occur only in start-ups. On the contrary, financial sector incumbents are responding in numerous ways to the technological disruption overtaking the sector. Moreover, technology companies, which have been historically close to consumers, are also emerging as providers of financial services. In the US, in 2016 alone, JP Morgan has spent more than $9.5 billion in revamping its IT infrastructure, out of which $600 million was spent on developing FinTech solutions, either in-house or through partnerships[2].

Despite significant investments in FinTech solutions, the literature to date has been limited in explaining what FinTech innovations are, where and why they emerge, and what is their impact on society and on the financial performance of companies that invest in them [2], in part due to a lack of a widely-accepted FinTech innovations taxonomy and datasets categorized according to such taxonomy. To fill in this gap, we aim to study the global landscape of FinTech innovations starting from a global patent dataset of over 100 million patent applications published between 2000 and 2017. Towards this goal, we first create a FinTech innovation taxonomy corroborated from an extensive literature search on working papers and published academic articles, reports and materials related to FinTech. We use a list of financial terms to pre-filter financial related patents. We then build a substantial corpus of manually labelled FinTech patents, and use it to train and evaluate different types of deep learning classifiers, with focus on BERT models [1].

The rest of the thesis is organized as follows: We discuss related work in Chapter 2. The FinTech dataset is described in Chapter 3, while the models we studied are introduced in Chapter 4. We describe our experimental setup in Chapter 5 and we discuss the results of the experiments in Chapter 6. Finally, we conclude the thesis and present ideas for future work in Chapter 7.

[2] https://www.jpmorganchase.com/content/dam/jpmc/jpmorgan-chase-and-co/investor-relations/documents/2016-annualreport.pdf

Chapter 2

Related works

Over the past decade, academic research in FinTech has grown in tandem with the exponential rise of new FinTech start-ups around the world [7]. Several journals have hosted special topics dedicated to FinTech Innovations, including the Review of Financial Studies [8] and the Journal of Management Information Systems [9]. However, very few studies have been able to provide a systematic overview of FinTech innovations, partly due to a lack of an internationally recognised taxonomy, and partly due to a lack of datasets that would enable large-scale analysis using machine learning and deep learning approaches. Such taxonomies and datasets are available for general innovations and have been used successfully to automatically classify patents and gain insights into general innovations trends in the last decade [10, 11, 12, 13, 14, 15, 16, 17, 18].

In the FinTech area, one of the first studies to use machine learning to identify FinTech innovations, and the implications on the financial performance of companies who invest in such innovations, was performed by Chen et al. [2]. The authors employed text-based machine learning approaches to classify and analyze innovations according to their key underlying technologies. Chen et al. [2] used a dataset of US patents covering years 2003-2017, pre-filtered using 487 financial terms. A subset of 1,800 patents was manually annotated according to 9 categories. These categories include 7 FinTech categories (specifically, *Cybersecurity*, *Mobile Transactions*, *Data Analytics*, *Blockchain*, *Peer-to-peer*, *Robo-adviser* and

Internet of Things), a category for financial patents that are not FinTech, and a category for non-financial patents. The manually annotated dataset was used to train and evaluate several machine learning classifiers. Empirical results showed that an ensemble classifier, consisting of linear support vector machines (SVM), Gaussian SVM, and neural network models trained on patent text, performed the best, with an accuracy of 82.6% and an F1 score of 76.3%.

In another recent study, Xu et al. [3] trained random forest (RF) classifiers (which can be seen as ensembles of decision trees) to identify FinTech patents. The original dataset used in their study was extracted from the Lens database, and covered years 2014-2018. A set of 478 financial terms was used to filter financial innovations. A subset of 1,800 patents was manually annotated according to 9 categories, including 7 FinTech categories (*Encryption & Security*, *Mobile Payments*, *Big Data Analytics*, *Blockchain*, *Online Lending*, *Expert Advisor*, and *Internet of Things*), and the 2 additional categories from [2]. The labeled subset was used to train and evaluate RF classifiers. Empirical results showed that the best performing classifier achieved an average accuracy of 71.67%.

	Dataset Characteristics	**Chen** *et al.* [2]	**Xu** *et al.* [3]	**Our dataset**
(1)	Source of patents	BDSS	Lens	**Orbis/Patsat**
(2)	Years covered	2003-2017	2014-2018	**2000-2017**
(3)	Legal jurisdiction of patents	US	US	**US + Europe**
(4)	IPC classes used	G&H	G&H	**G&H**
(5)	Initial number of patents based on criteria (1)-(4) above	200151	1181162	**6.8M**
(6)	Financial terms for filtering financial patents	487	478	**516**
(7)	Number of patents after filtering with financial terms (6)	67948	37156	**38228**
(8)	Number of FinTech categories considered	7	7	**5**
(9)	Number of manually annotated patents used for training	1800	1800	**1938+450**
(10)	Total number of FinTech patents identified out of (7)	6511	3602	**25580**

Table 2.1: *Characteristics of two existing FinTech datasets used by Chen et al. [2] and Xu et al. [3] by contrast with our dataset.*

While the datasets used by Chen et al. [2] and Xu et al. [3] do not include exactly the same categories, they are based on similar raw data (i.e., patent applications filed by inventors, in patent offices such as USPTO or EPO). The characteristics of the two datasets are summarized in Table 2.1, and some of them are discussed below:

- Both datasets consist of filtered patents with legal jurisdiction in the US, and belong to the G&H classes from the International Patent Classification (IPC) hierarchy.

- Both [2] and [3] used similar "lists of financial terms" consisting of 487 and 478 terms, respectively, to filter patents potentially related to financial services.

- Both studies identified 7 FinTech categories, and 2 additional categories to capture not FinTech, and non-financial patents, respectively. Chen et al. [2] identified the seven FinTech categories based on insights from a general reading of FinTech reports and articles. Xu et al. [3] selected their seven FinTech categories based on a Financial Stability Board (FSB) report from 2017.

- Both studies manually labeled small subsets of patents (specifically, 1,800 patents) according to the 9 categories considered. Chen et al. [2] labeled 200 patents in each of the 9 categories considered, while Xu et al. [3] selected a random sample of 1800 patents and labeled them according to the 9 categories.

- Both studies used only the *abstract* section of a patent.

The prior works on FinTech innovation classification [2, 3] have employed traditional machine learning approaches, and have found that ensemble-type approaches show promising results. However, in the light of the growing success that deep learning approaches have seen in recent years, several works [14, 16, 15, 17, 18] have used such approaches to automatically classify general patents according to standard categories in the International Patent Classification (IPC) or the Cooperative Patent Classification (CPC) taxonomies, and to improve the overall financial technology solutions.

For example, [13, 15] used recurrent neural networks, specifically, long short-term neural networks (LSTM) [19], together with word2vec embeddings [20], to classify patents into IPC categories, while [21] used gated recurrent unit (GRU) networks, together with fastText embeddings [22] for the same task. Similarly, [17, 14] used word embeddings, including Word2vec [20] and GloVe [23], together with convolutional neural networks (CNN) for text classification [24]. Hu et al. [16] build a hierarchical feature model that combined CNN and bidirectional LSTM (bi-LSTM) networks to capture both local lexical-level features and global sequential dependencies. The authors showed that the combined model achieved bet-

ter performance than the independent CNN and LSTM/Bi-LSTM models on mechanical patent documents. Finally, Lee and Hsiang [18] obtained state-of-the-art results with BERT models [1] on the task of classifying patent documents according to the IPC or CPC taxonomies. Specifically, [18] used large datasets of patent documents to fine-tune a pre-trained BERT-base model on the general patent classification task.

Beyond simply improving performance on the task of automatic patent classification, it is also of interest to analyze innovation trends, as cutting-edge technologies are permanently pioneered by scientists across the world. For example, trends within the technology domain, or within the financial services sector, can be foreseen by analyzing patent applications and patent grants. Chae and Gim [25] proposed a model based on the existing patent classification schemes, IPC and CPC, to extract accurate innovation trends, as well as common invention patterns. Sofean et al. [26] emphasized the need to predict proper relationships and trends among technological areas of inventions, and proposed to use natural language processing techniques to achieve this task. In the FinTech area, Chen *et al.* [2] used the results of their best ensemble classifier on a large set of financial patents to identify temporal trends with respect to the category of the innovation technology or with respect to its author. Furthermore, they used the results to estimate the value of a category by considering stock price responses, and showed that the *Internet of Things*, *Robo-advising*, and *Blockchain* categories generate the most significant financial gains to companies that invest in them.

To advance the research on identifying FinTech innovation started by Chen et al. [2] and Xu et al. [3], in this work, we first refine the list of financial terms provided by Chen et al. [2], and use them to identify patents, with jurisdiction in both the US and Europe, potentially related to financial services. Furthermore, we propose a new, improved FinTech taxonomy consisting of 5 categories (specifically, *Insurance*, *Payments*, *Investments*, *Fraud*, and *Data Analytics*), and manually label 2,530 according to these categories. In addition, we label 1,500 *Non-FinTech* patents. Given the success of the deep learning approaches on general patent classification tasks, and in particular, the state-of-the-art results produced by BERT models, we focus on classifying FinTech patents using BERT-type models and compare the top 10 models. Chen et al. [2] and Xu et al. [3], used only the *abstract* section

of a patent in their FinTech patent classification tasks. In addition to the *Abstract* section we are using also the *Title* and the *Claims* sections of a given patent. The goal is to compare the performance of these three sections and to observe what part of a patent section gives the highest results in the classification task.

Chapter 3

FinTech dataset

In this section, we describe the process we followed to create our FinTech dataset and taxonomy, as well as the annotation of the subset used for training the deep learning models.

Figure 3.1: *WordCloud of the 50 most frequent financial terms in the set of 38,228 patent documents used in our analysis. The size of a term is proportional with the number of patents in which the term appears.*

3.1 FinTech dataset construction

We retrieve all the patents filed between 2000 and 2017 from the matched Orbis-PATSTAT database, which contains bibliographical entries for over 100 million patent documents filed around the world, as well as details about their corporate ownership. From these, we select the patents developed/owned by corporate entities, and further filter only those with IPC codes pertaining to the fields G and H, which cover innovations related to digital computing, including many FinTech categories [2]. This filtering process results in 6.8 million patents. Starting with a dictionary of 487 financial terms from the Campbell R. Harvey's Hypertextual Finance Glossary and the online Oxford Dictionary of Finance and Banking [2], we develop an enhanced set of 516 financial terms, and use these terms to select patents that contain at least one keyword from the set. The number of patents that contain at least one of the 516 financial terms is 38,228. The distribution of the 50 most frequent financial terms in our dataset is illustrated in Figure. 3.1, where the size of each term is proportional to the number of patent documents in which that term appears. We use this resulting dataset of 38,228 potential FinTech patents in our analysis as outlined below. Characteristics of our dataset are summarized in the last column of Table 2.1, by contrast with the characteristics of the prior datasets [2, 3]. It is worth noting that our initial dataset is the largest among the three, as it covers both US and European patents.

3.2 FinTech innovation taxonomy

There is a wide range of financial products and services that fall under the FinTech umbrella. Currently, there is no comprehensive, well-accepted taxonomy to analyse the sector. Hence, we build a FinTech taxonomy by corroborating taxonomies which emerged from our research of numerous articles, reports and market maps from both academia and industry [33, 34, 35, 36, 37, 38, 2, 39]. Our taxonomy aims to capture innovations that pursue the integration of more sophisticated IT tools and data science solutions in financial products. It contains five FinTech categories, specifically, *Data Analytics*, *Fraud*, *Insurance*,

Investments and *Payments*. Applications corresponding to these categories, together with an example of a patent filling abstract in each category are shown in Table 3.1. Our FinTech taxonomy is aligned with that of [2]. However, in some respect, our taxonomy is more general as we include a broader range of FinTech innovations, but in other respects, we exclude some applications which are not necessarily specific to the financial sector, included [2] (e.g., *Blockchain* and *Internet-of-Things*).

3.3 FinTech dataset annotation

To be able to train machine learning and deep learning models for FinTech patent identification, we manually annotated/labeled a subset of our patent dataset. Specifically, we manually labeled the following number of patents: 440 patents for *Fraud*, 402 patents *Insurance*, 484 patents for *Investments*, 426 patents for *Payments*, and 186 patents in the *Data Analytics* category (the number of manually labeled patents in this category is smaller as these patents were more difficult to identify during the manual analysis). Furthermore, we manually labeled a subset of 450 Non-FinTech patents. Thus, together our manually labeled dataset contains 1.938 FinTech patents and 450 Non-FinTech patents, for a total of 2,388 manually labeled patents. To train and evaluate our models, the dataset was split into training and test subsets, where the training subset contains 80% of the labeled data and the test dataset contains 20% of the data.

Fintech category	Applications	Examples of patent filing abstracts
Data Analytics	**Software, data infrastructure and analytics for financial services**	"[. . .] computer programs product are provided for automated generic and parallel aggregation of characteristics and key figures of unsorted mass data being of specific economic interest, particularly associated with financial institutions, and with financial affairs in banking practice."[27]
Fraud	**Fraud detection, security infrastructure, identity verification & compliance**	"This invention provides a system and method for reducing the fraud related to remittance transactions initiated at web portals. [..] For example, a funding agency computer that enables a remittance transaction can request that a mobile platform computer verify a customer with a mobile personal identifier. The mobile platform computer can request the mobile personal identifier from a customer via the customer's mobile handset device."[28]
Insurance	**Life insurance, general & (re)insurance software analytics**	"An automated assignment system may operate with a computer to automatically assign insurable events to one or more organizational entities associated with an insurance organization. The automated assignment system may categorize the insurable event."[29]
Investments	**Portfolio management, lending and investing platforms and portfolio analytics**	"A visual interactive multi-criteria decision-making method and computer-based apparatus for portfolio management. The method/apparatus supports partitioning of a portfolio of physical or other assets into two mutually exclusive categories, such as assets recommended for sale and assets recommended for retention."[30]
Payments	**Mobile payments**	"A mobile payment platform and service provides a fast, easy way to make payments by users of mobile devices. The platform also interfaces with non-mobile channels and devices such as e-mail, instant messenger, and Web. In an implementation, funds are accessed from an account holder's mobile device such as a mobile phone or a personal digital assistant to make or receive payments."[31]
	Digital wallets	"A system and a method are provided for generating a digital receipt for purchases made utilizing a digital wallet or with other payment procedures. The digital receipt is stored in the cloud in a digital receipts repository for later retrieval. The digital receipt can be standardized to facilitate data processing of the data contained in data fields of the digital receipt."[32]

Table 3.1: *Proposed FinTech Taxonomy (which includes five categories). Applications corresponding to each categories are also shown, together with an example of a patent filing abstract in each category.*

Chapter 4

BERT-based models

In this section, we describe the deep learning models that we use in our analysis of FinTech patents.

4.1 BERT model

We use BERT, which stands for Bidirectional Encoder Representations from Transformers [1], as the core approach for the task of classifying FinTech patent documents, given that BERT models have produced state-of-the-art results for many text classification tasks [40], including classification of general patent documents [18]. BERT is a language model that uses a deep bidirectional transformer encoder architecture [41] to encode sentences and their tokens into dense vector representations. A generic model is pre-trained on a large corpus of un-annotated text (e.g., Wikipedia) using two self-supervised learning tasks: masked word prediction (a.k.a., masked language modeling, or MLM) and next sentence prediction (NSP). BERT takes as input a sequence of word tokens, where the first token is a special token denoted by [CLS] (the output representation of the [CLS] token can be seen as a semantic representation for the whole input sequence). A BERT input sequence consists of one or two sentences. For an input sequence consisting of two sentences, the two sentences are separated by another special token, denoted by [SEP]. Embeddings of the input tokens are

provided to a multi-layer bidirectional transformer encoder, which transforms the original input embeddings into contextual output embeddings using the masked word prediction and/or next sentence prediction tasks. Figure 4.1 shows the architecture of a generic BERT model, which takes two sentences as input.

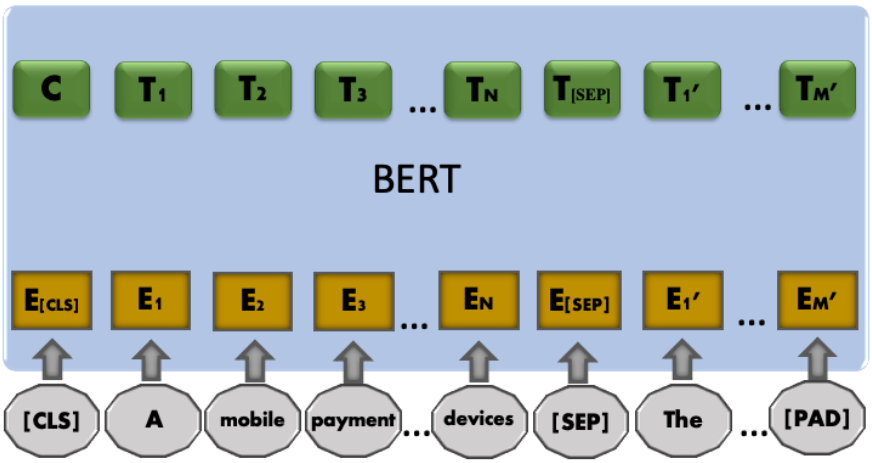

Figure 4.1: *BERT architecture used for pre-training (figure adapted from [1]).*

A generic BERT model, pre-trained on a large corpus, can be further *pre-trained* and/or *fine-tuned* for specific NLP tasks [1]. Particularly, a BERT model for the FinTech patent classification task can be initialized with the parameters of a generic pre-trained BERT model, further pre-trained using FinTech patent data, and subsequently fine-tuned for patent classification. In the pre-training phase, the input to the model consists of tokens in two sentences (from patent text), preceded by the token [CLS] and separated by the token [SEP], and the output consists of vector representations of the input tokens. In the fine-tuning phase, the BERT architecture is similar to the architecture of a generic BERT model, except for the inclusion of a classification component (e.g., a fully connected layer, followed by a softmax classification layer). The classification component is linked to the first output embedding, C, corresponding to the first input token [CLS], and provides a representation of the whole input sequence. The input to the classification BERT model consists of tokens

in a patent text (also preceded by [CLS]), and the output is the category of the input patent.

4.2 BERT variants

The success of the initial BERT-based models has resulted in an unparalleled suite of variants that can be used with the pre-training/fine-tuning framework proposed in [1]. We used six variants in our analysis, specifically, RoBERTa [42], ALBERT [43], XLNet [44], BART [45], Longformer [46] and DeBERTa [47]. RoBERTa (Robustly Optimized BERT Approach) [42] uses a larger dataset and an improved procedure to pre-train the BERT architecture. Among others, the next sentence prediction is removed and the masking applied to the training data is changed dynamically. ALBERT (A Lite BERT) [43] is focused on decreasing BERT's size (i.e., the number of parameters that need to be learned), while not hurting its performance. It achieves a reduction in the number of parameters by factorizing the embedding parameterization and sharing parameters across all layers. To improve the training, it replaces the next sentence prediction task with a sentence order prediction task that better captures the inter-sentence cohesion. XLNet [44] is a large bidirectional transformer, whose authors argue against the masked language modeling task and introduce an autoregressive permutation language modeling task for training (specifically, prediction of the next token in a sequence using some random order of the sequence). This improvement in the training procedure enables XLNet to capture better bidirectional dependencies among tokens in a sequence. BART[45] is a sequence-to-sequence pre-trained model which unifies concepts from BERT and GPT-2 [48] architectures. More specifically is using a bidirectional encoder but also a left-to-right decoder. BART is pre-trained on tasks where as input is given a text. On this text a noising function is applied randomly. The goal of Bart is to apply the denosing auto-encoder mechanism to reconstruct the original text. The authors are mentioning that on some language modeling tasks the performance of Bart matches RoBERTa. However BART achieves state-of-the-art on other NLP tasks as question answering, summarizing or abstractive dialogue. Longformer [46] includes an improved attention mechanism pattern which makes this model very suitable for language modeling tasks of very long doc-

uments. If the majority of BERT-like models supports sequences length up to 512 tokens, Longformer can handle up to 4,096 tokens. A key feature in Longformer architecture is that the time and memory complexity will scale linearly with the sequence length, while the standard BERT-like models self-attention mechanism grow quadratically with the sequence length. DeBERTa [47] is one of the newest pre-trained models, which improves BERT and RoBERTa performance by using two new approaches: a disentangled attention mechanism and an enhanced mask decoder.

Pre-trained model	**Architecture name**	**Details of the pre-trained model**
BERT Oct 2018	bert-base-uncased	12-layer, 768-hidden, 12-heads, 110M parameters. Trained on lower-cased English text.
	bert-large-uncased	24-layer, 1024-hidden, 16-heads, 336M parameters. Trained on lower-cased English text.
	bert-base-cased	12-layer, 768-hidden, 12-heads, 109M parameters. Trained on cased English text.
	bert-large-cased	24-layer, 1024-hidden, 16-heads, 335M parameters. Trained on cased English text.
	bert-large-uncased whole-word-masking	24-layer, 1024-hidden, 16-heads, 335M parameters. Trained on lower-cased English text using Whole-Word-Masking
	bert-large-cased whole-word-masking	24-layer, 1024-hidden, 16-heads, 335M parameters. Trained on cased English text using Whole-Word-Masking
XLNet Jun 2019	xlnet-base-cased	12-layer, 768-hidden, 12-heads, 110M parameters. XLNet English model
	xlnet-large-cased	24-layer, 1024-hidden, 16-heads, 340M parameters. XLNet Large English model
RoBERTa Jul 2019	roberta-base	12-layer, 768-hidden, 12-heads, 125M parameters RoBERTa using the BERT-base architecture
	roberta-large	24-layer, 1024-hidden, 16-heads, 355M parameters RoBERTa using the BERT-large architecture
ALBERT Sep 2019	albert-base-v1	12 repeating layers, 128 embedding, 768-hidden, 12-heads, 11M parameters
Bart Oct 2019	bart-base	12-layer, 768-hidden, 16-heads, 139M parameters
Longformer Apr 2020	longformer-base	12-layer, 768-hidden, 12-heads, ~149M parameters Starting from RoBERTa-base checkpoint, trained on documents of max length 4,096
DeBERTa Jun 2020	deberta-base	12-layer, 768-hidden, 12-heads, ~125M parameters DeBERTa using the BERT-base architecture

Table 4.1: *BERT-like pre-trained models used in this thesis*

Chapter 5

Experimental setup

We investigate different BERT like models, for FinTech patent classification. The goal is to understand how the results vary with the BERT models used and what model performs the best on three patent sections: title, abstract and claims and combinations of this sections. In what follows, we provide details about the implementation and hyper-parameters used for the models that we experiment with.

5.1 Implementation details

Using the Transformers library by HuggingFace [49], we experiment with 14 pre-trained BERT models and variants, by fine-tuning the models using labeled data for our specific FinTech classification task. The models we experiment with include BERT, RoBERTa, XLNet, ALBERT, BART, Longoformer and DeBERTa. We use different architectures for each of these models (e.g., for BERT we used architectures such as: *bert-base-uncased*, *bert-base-cased*, *bert-large-uncased*, etc.). We train each model for 2 epochs, using the AdamW optimizer with a learning rate of $2e^{-5}$. We use default values for other hyper-parameters.

5.2 Evaluation metrics

To evaluate the performance of the various models that we train, we use several standard metrics, including the overall accuracy, precision, recall and F1 scores. We also report the precision, recall, and F1 scores for each of our categories to determine what categories might be easier or harder to identify.

Chapter 6

Results and discussion

In this chapter, we present the results of the deep learning models' ability to correctly classify FinTech patents into different categories (*Insurance, Payments, Investment, Fraud, Data Anayltics, Non-FinTech*). We fine-tune the BERT-based models on our labeled training dataset, and estimate their performance on the test dataset of patent documents.

Our patent documents are sectioned into segments of *title, abstract,* and *claim*, and our models use these discrete segments to classify the documents. Classification was performed using one section of a patent document, as well as using a concatenation of the segments.

The results of our classifiers predictions for examining a segment of a patent document will be presented in three steps:

- First, we report the F1-scores performance for the 10 pre-trained best-performing models.
- We will then provide numeric result for the top 5 pre-trained models performance in terms of precision (Pr), recall (Re) and F1-score (F1), for both overall performance as well as for each category of the FinTech taxonomy.
- Finally, we show the confusion matrix that corresponds to the best model per each section of a patent.

6.1 Model prediction performance on the *title* section of patent document

This section reports on how well classifiers would predict FinTech patents based solely on title. In our dataset, the longest title has 36 tokens and the smallest title has only 2 tokens (e.g., *Fraud detection* or *Payment system*). Given the reduced number of tokens, the models have more challenges in learning from a limited span of tokens, which in some cases are not very informative (e.g., a title named *Data excavator* or *Hypothetical-portfolio-return determination*).

6.2 Analysis of the 10 best performing models by F1-score on the title section

A comparison of the 10 best-performing models for the *title* section is shown in Figure 6.1. The F1-scores range from 78% (for *deberta-base*) to 82.1% (for *roberta-large*), showing that the BERT-like models are generally decent for classifying the patent-titles of our FinTech classification problem.

The best performing model, *roberta-large*, has 24-layers, 1024-hidden units, 16-heads and 3550M parameters, and is pre-trained on 160GB of English text.

The next two models (having similar scores to each other 80.8% and 80%) are *bert-large-uncased-whole-word-masking, bart-base*. Is important to mention that from the top 4 models, only the *bart-base* is a cased models, while the others models are uncased.

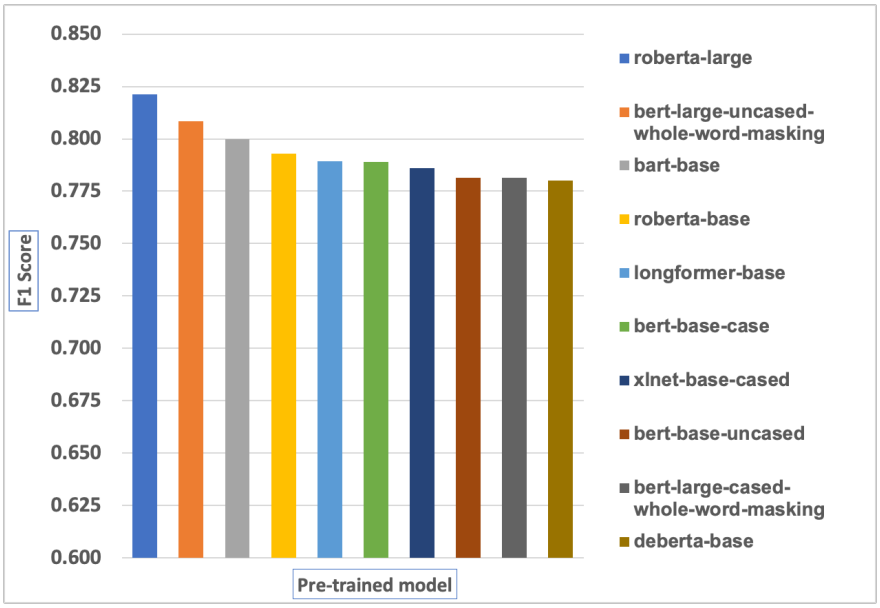

Figure 6.1: *Comparison of the 10 best-performing BERT-like models in terms of F1-Score for the* `Title` *section. The* `roberta-large` *model has the best performance overall.*

6.3 Numeric results for the top 5 performing models on the *title* section

Table 6.1 shows a greater articulation of results by showing precision (Pr), recall (Re) and F1-score (F1) for each FinTech category for the top five performing models. In terms of individual categories, the results show that the *title* segment had best performance for *Investment* category, with an F1-score of 92.32%, while *Data Analytics* category has the worst performance using *title*, with a F1-score of 56.00%. To some extent, the results may reflect the imbalance from our dataset distribution, with, for instance, *Data Analytics* underrepresented in our sample. While the *Investment* category has the most patents/titles 484, *Data Analytics* is the category represented only by 186 patents/titles.

Category	Metric	**roberta large**	**bert large uncased whole word masking**	**bart base**	**roberta base**	**longformer base**
Insurance	Pr	**0.932**	0.882	0.912	0.914	0.805
	Re	0.84	0.827	0.765	0.79	**0.864**
	F1	**0.883**	0.854	0.832	0.848	0.833
Payments	Pr	0.774	0.755	**0.867**	0.747	0.847
	Re	0.847	**0.906**	0.765	0.835	0.718
	F1	0.809	**0.824**	0.812	0.789	0.777
Investment	Pr	0.918	0.869	0.879	0.804	**0.935**
	Re	**0.928***	0.887	0.897	**0.928***	0.887
	F1	**0.923**	0.878	0.888	0.861	0.91
Fraud	Pr	**0.868**	0.807	0.643	0.768	0.679
	Re	0.67	0.761	**0.818***	0.716	**0.818***
	F1	0.756	**0.784**	0.72	0.741	0.742
Data Analytics	Pr	0.553	0.654	0.613	0.613	**0.765**
	Re	**0.568**	0.459	0.514	0.514	0.351
	F1	0.56	0.54	**0.559***	**0.559***	0.481
Non-FinTech	Pr	0.769	0.804	0.828	**0.83**	0.75
	Re	**0.922**	0.822	0.856	0.811	0.867
	F1	0.838	0.813	**0.842**	0.82	0.804
Average	Pr	**0.829**	0.811	0.809	0.796	0.802
	Re	**0.822**	0.812	0.799	0.795	0.795
	F1	**0.821**	0.808	0.8	0.793	0.789
	Acc	**0.822**	0.812	0.799	0.795	0.795

Table 6.1: *Performance results on the test data only for the* `Title` *section. The best results for the top 5 deep learning models, are highlighted in boldface. With boldface and * are marked those situations when different models had same results (per category and metric)*

6.4 Confusion matrix for the best model on the *title* section

Figure 6.2 shows the confusion matrix corresponding to the "robert-large," the best model for the *title* section. The diagonal entries show the percentage of correctly classified instances in each category, while the non-diagonal entries show how the misclassified instances for a particular category are distributed among the other categories.

As can be seen, the *Investment*, *Non-Fintech*, *Payments* and *Insurance* have a high percentage of instances correctly classified (93%, 92%, 85% and 84%). For *Data Analytics*, 57% of instances are classified correctly, while 43% are misclassified as: *Non-Fintech* (22%), *Payments* (8%) , *Insurance* and *Fraud* with (5%) and *Investment* (3%).

Figure 6.2: *Normalized confusion matrix corresponding to the best model for the* `Title` *section is* `roberta-large` . *The diagonal entries show the percentage of correctly classified instances in each category. Non-diagonal entries on a row show how the misclassified instances of a category are distributed among the other categories.*

6.5 Model prediction performance on the *abstract* section of patent document

Compared to *title* and *claims*, the *abstract* section of a patent represents a brief non-technical description of the FinTech invention. The abstract examples present in our dataset have between 19 and 540 tokens with an average of 150 tokens. As expected, the *abstract* is longer than the *title*, and this fact will have a correlation in the performance results of the pre-trained models. Given that the *abstract* section gives a longer text sequence an input to the pre-trained models compared to *title* section, we observed, as hypothesized, that the pre-trained models improved in performance over *title*.

6.6 Analysis of the best 10 performing models by F1-score on the *abstract section*

A comparison of the 10 best-performing models for the *abstract* section is shown in Figure 6.3. F1-scores ranges from 95.93% (for *deberta-base*), to 96.90% (for *bert-large uncased-whole-word-masking*) showing a substantial increase from the F1-scores of the *title* section, where the range was between 78% and 82%.

As expected the *abstract* section gives a better contextual information of the FinTech invention than *title*, therefore the pre-trained models have a better prediction results.

The top 10 best pre-trained models are presented in Figure 6.3, where *bert-large-uncased-whole-word-masking* has the highest F1-score: 96.90%. Following closely is the *bert-large-cased* and *bert-base-cased* with 96.70% and 96.60% respectively.

While *bert-large-uncased-whole-word-masking* and *bert-large-cased* share the same architecture, specifically, 24-layers,1024-hidden units, 16-heads, and 335M parameters, *bert-base-cased* has only 12-layers, 768-hidden units, 12-heads and 109M parameters. Hence a more complex architecture is not substantially increasing the F1-score.

A key distinction between these three models is the masking method used. More specifi-

cally, *bert-large-uncased-whole-word-masking* model, as the name suggests, is masking *whole words*, which helps in performance gain while *bert-large-cased* as well as *bert-base-cased* is masking random word pieces.

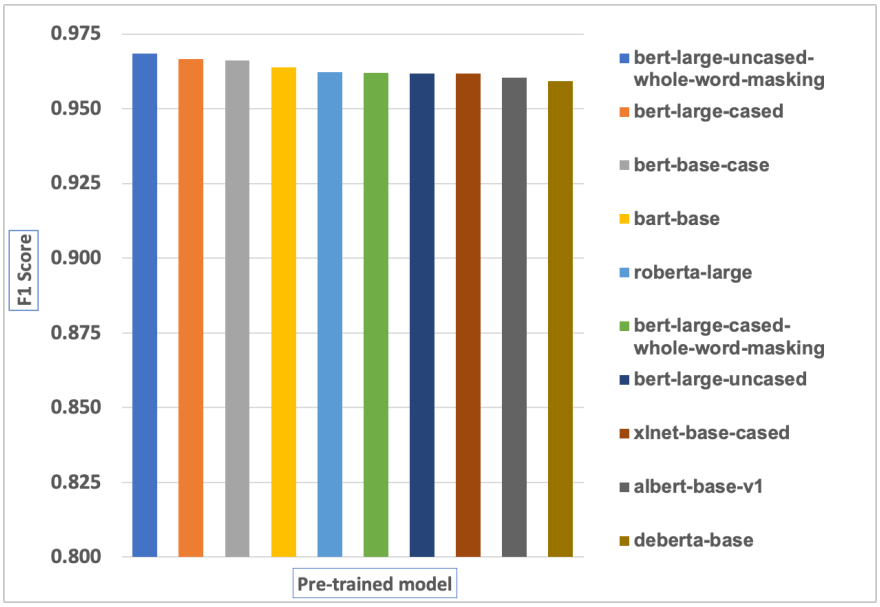

Figure 6.3: *Comparison of the 10 best-performing BERT-like models in terms of F1-Score for the* `Abstract` *section. The* `bert-large-uncased-whole-word-masking` *model has the best performance overall.*

6.7 Numeric results for the top 5 performing models of the *abstract* section

In Table 6.2, we show the results for each category separately, and also the average over the 6 categories captured by our labeled data (including 5 FinTech categories and 1 Non-FinTech category), and the accuracy of the models. We can observe that all 5 models have similar results, with *bert large uncased whole word masking* having the best average scores of 96.90% for precision(Pr), recall(Re), F1-score(F1) and accuracy(Acc). In terms of individual categories, the results show that the *Fraud* category has the best performance, with an F1-score of 98.90%, while the *Data Analytics* category has the worst performance,

with an F1-score of 87.70%.

In contrast with the results from the *title* section, we notice the most substantial improvement is for *Data Analytics*, from 56% to 87.70% F1-score. This is a strong evidence that the *abstract* section is an improvement on *title* in detecting FinTech innovations.

Category	Metric	**bert large uncased whole word masking**	**bert large cased**	**bert base cased**	**bart base**	**roberta large**
Insurance	Pr	0.976	**0.988**	0.976	0.953	0.964
	Re	**1***	0.988	**1***	**1***	**1***
	F1	**0.988***	**0.988***	**0.988***	0.976	0.982
Payments	Pr	0.943	0.932	0.953	0.943	**0.965**
	Re	**0.976***	0.965	0.965	**0.976***	0.965
	F1	0.96	0.948	0.959	0.96	**0.965***
Investment	Pr	**0.979**	0.969	0.969	0.96	0.978
	Re	0.959	**0.979***	0.969	**0.979***	0.928
	F1	0.969	**0.974***	0.969	0.969	0.952
Fraud	Pr	**0.978***	**0.978***	**0.978***	**0.978***	**0.978***
	Re	**1***	**1***	**1***	**1***	**1***
	F1	**0.989***	**0.989***	**0.989***	**0.989***	**0.989***
Data Analytics	Pr	0.889	0.892	0.912	**0.938**	0.868
	Re	0.865	**0.892**	0.838	0.811	**0.892***
	F1	0.877	**0.892**	0.873	0.87	0.88
Non-FinTech	Pr	**1***	**1***	0.977	**1***	0.966
	Re	**0.956***	0.933	**0.956***	0.933	**0.956***
	F1	**0.977**	0.966	0.966	0.966	0.961
Average	Pr	**0.969**	0.967	0.966	0.965	0.963
	Re	**0.969**	0.967	0.967	0.964	0.962
	F1	**0.969**	0.967	0.966	0.964	0.962
	Acc	**0.969**	0.967	0.967	0.964	0.962

Table 6.2: *Performance results on the test data only for the* `Abstract` *section. The best results for the top 5 deep learning models, are highlighted in boldface. With boldface and * are marked those situations when different models had same results (per category and metric). The best model for the* `Abstract` *section is* `bert-large-uncased-whole-word-masking`.

6.8 Confusion matrix for the best model on the *abstract* section

Figure 6.4 shows the confusion matrix corresponding to the *bert large uncased whole word masking* the best model for the *abstract* section. The diagonal entries show the percentage of correctly classified instances in each category, while the non-diagonal entries show how the misclassified instances for a particular category are distributed among the other categories. As can be seen, the *Insurance*, *Payments*, *Investment* , *Fraud* and *Non-Fintech* have a high percentages above 96.00%. For *Data Analytics*, 86% of instances are classified correctly, while 14% are misclassified.

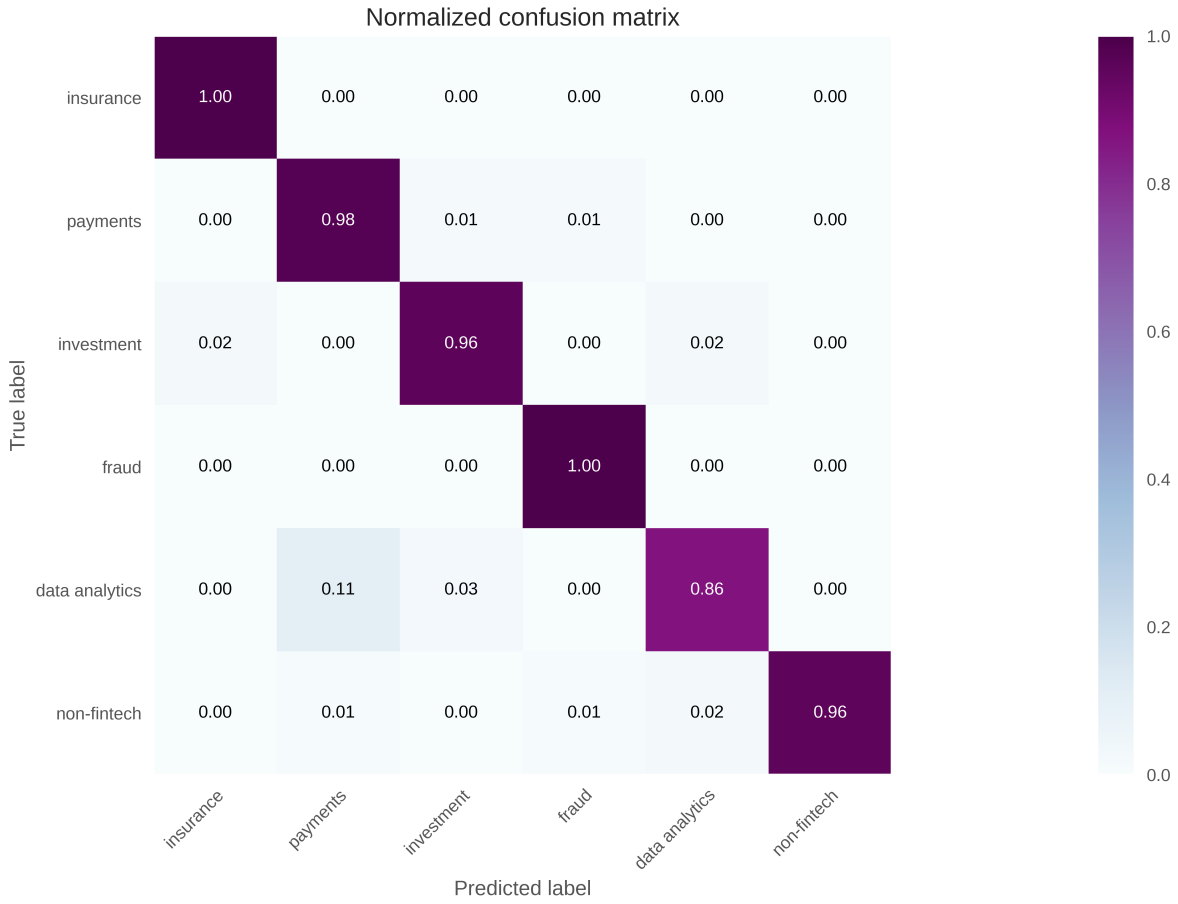

Figure 6.4: *Normalized confusion matrix corresponding to the best model for the* `Abstract` *section :* `bert-large uncased`*. The diagonal entries show the percentage of correctly classified instances in each category. Non-diagonal entries on a row show how the misclassified instances of a category are distributed among the other categories.*

6.9 Model prediction performance on the *claims* section of patent document

The claims section of a patent is usually substantially longer than the abstract section, with an minimum token content of 40, average 1,500, with a maximum of 11,000. Claims might be considered the heart of a patent, from a legal point of view, and employ specialized language.

6.10 Analysis of the best 10 performing models by F1-score on the *claims* section

F1-scores ranges from 89.96% (for "xlnet-base-cased"), to 91.62% (for "bert-large-cased-whole-word-masking") showing a substantial increase from the F1-scores of the *title* section, where the range was between 78% and 82%, but a slightly decrease from the F1-scores of the *abstract* section, where the range was 95.93% to 96.90%.

The top 10 best pre-trained models are presented in Figure 6.5, where "bert-large-cased-whole-word-masking" has the highest F1-score: 91.62%. Following closely is the "bert-large-cased" and "bart-base" with 91.42% and 91.17% respectively.

While "bert-large-cased-whole-word-masking" and 'bert-large-cased" share the same architecture, specifically, 24-layers,1024-hidden units, 16-heads, and 335M parameters, 'bart-base" has only 12-layers, 768-hidden units, 16-heads and 139M parameters. Hence, the top three share the same number of heads, but different numbers of layers and parameters.

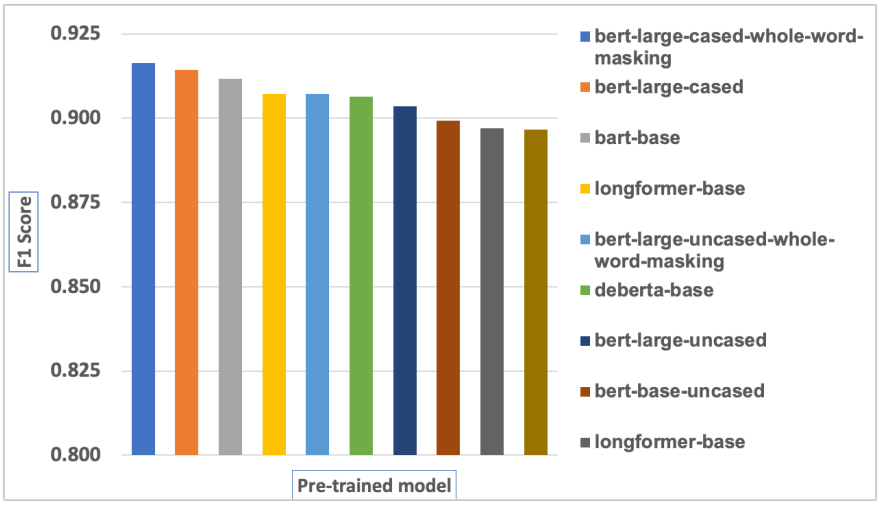

Figure 6.5: *Comparison of the 10 best-performing BERT-like models in terms of F1-Score for the* `Claims` *section. The* `bert-large-cased-whole-word-masking` *model has the best performance overall.*

6.11 Numeric results for the top 5 performing models of the *claims* section

In Table 6.3, we show the results for each category separately, and also the average over the 6 categories captured by our labeled data (including 5 FinTech categories and 1 Non-FinTech category), and the accuracy of the models.

We can observe that all 5 models have similar results, with "bert large cased whole word masking" having the best average scores of 91.60% for recall (Re), F1-score(F1), accuracy (Acc) and 91.80% for precision (Pr).

In terms of individual categories, the results show that the *Insurance* has an overall F1-score higher than rest of the FinTech categories, while the *Data Analytics* category has the worst performance, with an average F1-score of 80%.

In contrast with the results from the *title*, we notice the most improvement is for *Data Analytics*, from 56% to 80% F1-score. However, *claims* is not as effective for *Data Analytics* as the *abstract*. Overall *Data Analytics*, which remains relatively low over all three individual sections.

Category	Metric	bert large cased whole word masking	bert large cased	bart base	longformer base	bert large uncased whole word masking
Insurance	Pr	0.962	**1**	0.94	0.952	0.963
	Re	0.951	0.938	0.975	**0.988**	0.975
	F1	0.957	0.968	0.958	**0.97**	0.969
Payments	Pr	0.835	0.844	**0.859**	0.844	0.856
	Re	**0.953***	**0.953***	0.929	**0.953***	0.906
	F1	0.89	**0.895***	0.893	**0.895***	0.88
Investment	Pr	0.939	0.921	**0.968**	0.957	0.938
	Re	0.948	**0.959**	0.928	0.928	0.938
	F1	0.944	0.939	**0.947**	0.942	0.938
Fraud	Pr	0.892	**0.893**	0.832	0.857	0.882
	Re	0.841	0.852	**0.898**	0.886	0.852
	F1	0.865	**0.872***	0.863	**0.872***	0.867
Data Analytics	Pr	0.909	0.853	0.929	0.92	**0.931**
	Re	**0.811**	0.784	0.703	0.622	0.73
	F1	**0.857**	0.817	0.8	0.742	0.818
Non-FinTech	Pr	**0.966***	0.954	0.965	**0.966***	0.895
	Re	0.933	0.922	0.922	**0.944***	**0.944***
	F1	0.949	0.938	0.943	**0.955**	0.919
Average	Pr	**0.918***	0.916	0.915	0.917	0.909
	Re	**0.916**	0.914	0.912	0.914	0.908
	F1	**0.916**	0.914	0.912	0.912	0.907
	Acc	**0.916**	0.914	0.912	0.914	0.908

Table 6.3: *Performance results on the test data only for the* `Claims` *section. The performance is reported in terms of precision (Pr), recall (Re) and F1-score (F1), for each category and overall. Furthermore, the overall accuracy (Acc) is also shown at the end. The best results for the top 5 deep learning models, are highlighted in boldface. With boldface and * are marked those situations when different models had same results (per category and metric). The best model for the* `Claims` *section is* `bert-large-cased-whole-word masking`.

6.12 Confusion matrix for the best model on the claims section

Figure 6.6 shows the confusion matrix corresponding to the *bert large cased whole word masking* the best model for the *claims* section. The diagonal entries show the percentage of correctly classified instances in each category, while the non-diagonal entries show how the misclassified instances for a particular category are distributed among the other categories.As can be seen, the *Insurance*, *Payments*, *Investment* , *Fraud* have predictions percentages of 95.00%. For *Data Analytics*, 81% of instances are classified correctly, while 19% are misclassified as: 5% *Payments*, *Investment* and *Fraud* and 3% as *Non-FinTech*.

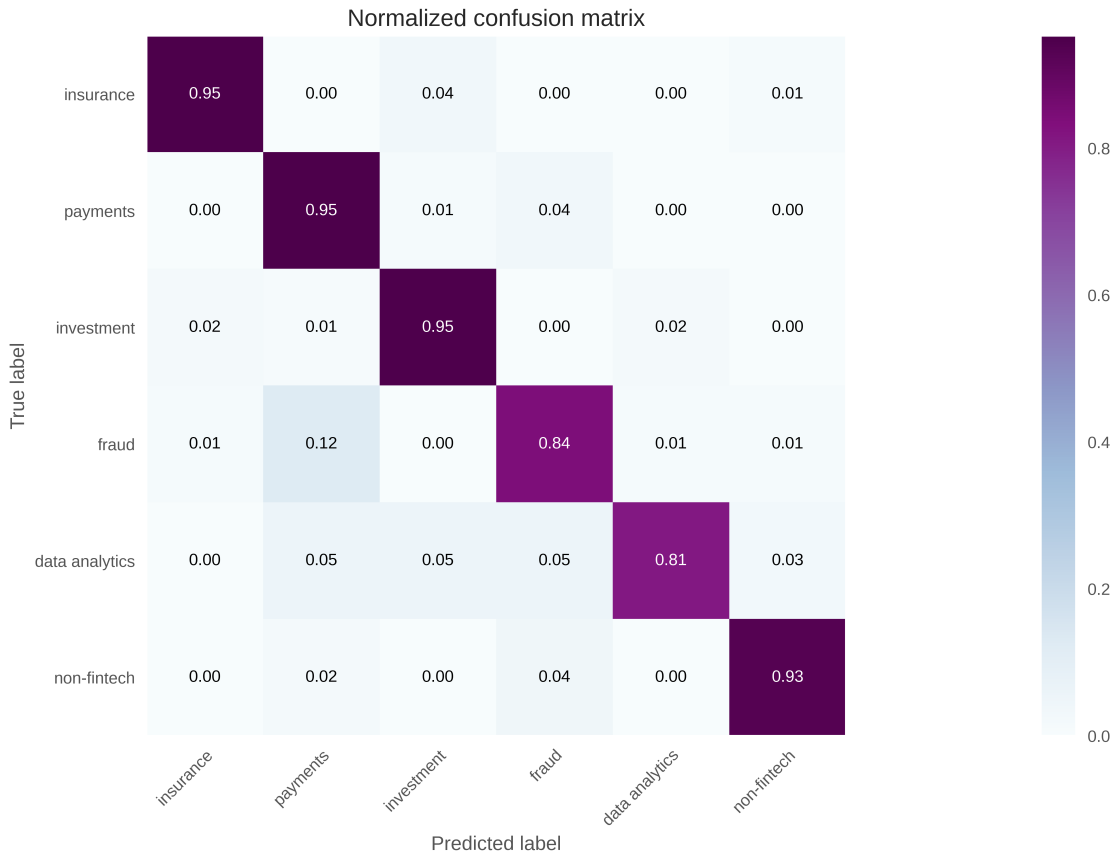

Figure 6.6: *Normalized confusion matrix corresponding to the best model for the* `Claims` *section :* `bert-large cased-whole-word-masking`. *The diagonal entries show the percentage of correctly classified instances in each category. Non-diagonal entries on a row show how the misclassified instances of a category are distributed among the other categories.*

6.13 Model prediction performance on using combined patent sections

In this section, we report the results of combinations of patent document sections in order to see which combinations yield the best performance. Specifically, we took the best performing model for each combination of *title+abstract, abstract+claims, title+claims,* and *title+abstract+claims*, and compared them for performance.

Table 6.4 shows that the best performing combination was *title+abstract*, with *abtract+claims* performing very closely. Worst performing was *title+claims*, which mirrors our individual section results, where *title* and *claims* were individually the worst performers. Overall, it was the *abstract* section combinations that provided the strongest synergy.

	Section	**tile+abstract**	**abstract+claims**	**title+abstract+claims**	**title+claims**
Category	Metric	xlnet-base-cased	xlnet-base-cased	longformer-base	longformer-base
Insurance	Pr	0.964	**0.976***	**0.976***	0.963
	Re	**1***	**1***	0.988	0.975
	F1	0.982	**0.988***	0.982	0.969
Payments	Pr	**0.954***	0.943	0.943	0.845
	Re	**0.976***	**0.976***	**0.976***	0.965
	F1	**0.965***	0.96	0.96	0.901
Investment	Pr	**0.979***	**0.979***	0.959	0.948
	Re	**0.969***	**0.969***	**0.969***	0.938
	F1	**0.974***	**0.974***	0.964	0.943
Fraud	Pr	**0.978***	**0.978***	**0.978***	0.918
	Re	**1***	0.989	**1***	0.886
	F1	**0.989***	0.983	**0.989***	0.902
Data Analytics	Pr	**0.969***	0.941	0.889	0.903
	Re	0.838	**0.865***	**0.865***	0.757
	F1	0.899	**0.901***	0.877	0.824
Non-FinTech	Pr	**0.989***	**0.989***	1	0.966
	Re	**0.978**	0.967	0.933	0.933
	F1	**0.983**	0.978	0.966	0.949
Average	Pr	**0.973**	0.971	0.965	0.927
	Re	**0.973**	0.971	0.964	0.925
	F1	**0.972**	0.97	0.964	0.924
	Acc	**0.973**	0.971	0.964	0.925

Table 6.4: *Results from best performing models on the test data for different patent document section combinations. The performance is reported in terms of precision (Pr), recall (Re) and F1-score (F1), for each category and overall. Furthermore, the overall accuracy (Acc) is also shown at the end. The best results for the top 4 deep learning models, are highlighted in boldface. With boldface and * are marked those situations when different models had same results (per category and metric).*

6.14 Confusion matrix for the best performing combination/model

Figure 6.7 shows the confusion matrix corresponding to the *xlnet-base-cased* the best model for the combination *title+abstract* section. The diagonal entries show the percentage of correctly classified instances in each category, while the non-diagonal entries show how the misclassified instances for a particular category are distributed among the other categories.

As can be seen, the *Insurance* and *Fraud* have perfect predictions, with a percentages of 100%. For *Data Analytics*, 84% of instances are classified correctly, while 16% are misclassified as: 3% *Insurance*, *Investment* and *Non-Fintech* and 8% as *Payments*.

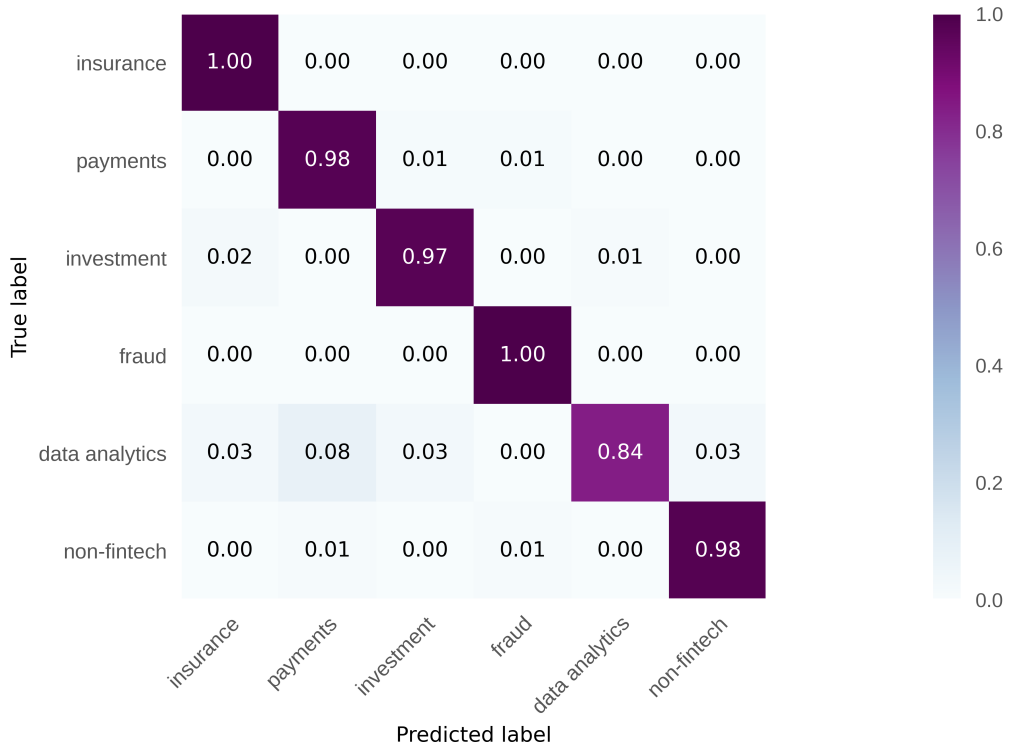

Figure 6.7: *Normalized confusion matrix corresponding to the best performing combination and model :* `title+abstract / xlnet-base-cased`. *The diagonal entries show the percentage of correctly classified instances in each category. Non-diagonal entries on a row show how the misclassified instances of a category are distributed among the other categories.*

6.15 Overall best models performance

As seen in Table 6.5 we present our overall best models for each individual patent section but also for the combined patent section.

We observe that the *title* section itself, has relatively poor perfromance compared with the *abstract* or *claims* section. However, when we concatenate *title + abstract* sections the pre-trained model *xlnet base cased* achives the highest overall performance in terms of F1-score, specifically 97.20% and an accuracy of 97.30%.

	Section(s)	**tile + abstract**	**abstract**	**claims**	**title**
Category	Metric	xlnet base cased	bert large uncased whole word masking	bert base uncased	roberta large
Insurance	Pr	0.964	**0.976**	0.951	0.932
	Re	0.954	**1**	0.951	0.84
	F1	0.979	**0.988**	0.951	0.883
Payments	Pr	**0.978**	0.943	0.857	0.774
	Re	0.969	**0.976**	0.918	0.847
	F1	**0.989**	0.96	0.886	0.809
Investment	Pr	0.973	**0.979**	0.891	0.918
	Re	**1**	0.959	0.928	0.928
	F1	**0.976**	0.969	0.909	0.923
Fraud	Pr	0.969	**0.978**	0.862	0.868
	Re	**1***	**1***	0.852	0.67
	F1	0.838	**0.989***	0.857	0.756
Data Analytics	Pr	**0.978**	0.889	0.818	0.553
	Re	**0.973**	0.865	0.73	0.568
	F1	**0.982**	0.877	0.771	0.56
Non-FinTech	Pr	0.965	**1**	0.976	0.769
	Re	**0.974**	0.956	0.922	0.922
	F1	**0.989**	0.977	0.949	0.838
Average	Pr	0.899	**0.969**	0.9	0.829
	Re	**0.983**	0.969	0.9	0.822
	F1	**0.972**	0.969	0.899	0.821
	Acc	**0.973**	0.969	0.9	0.822

Table 6.5: *Comparison between the best models for each individual patent section and the best model for the combined sections. The performance is reported in terms of precision (Pr), recall (Re) and F1-score (F1), for each category and overall. Furthermore, the overall accuracy (Acc) is also shown at the end. With boldface and * are marked those situations when different models had same results (per category and metric).*

Chapter 7

Conclusions and future work

We develop a FinTech taxonomy, and label a dataset according to this taxonomy to enable studies of FinTech innovations. We train BERT-based models to identify FinTech patents, and use them to shortlist a set of 25,580 FinTech patents in one of the following categories: *Data Analytics*, *Fraud*, *Insurance*, *Investments* and *Payments*.

We conclude that complementary information across title, abstract, and claims section leads to better category prediction. To our knowledge, our research could be the most accurate description of FinTech innovation to date, and has wider overall implications for FinTech patent classification and wider research in innovation and trends.

As part of future work, we plan to further improve performance by using domain adaptation and transfer learning approaches, which can benefit from general patent data, in addition to FinTech data.

Finally, we believe that our taxonomy and dataset will help to substantially reduce the search costs for FinTech innovations, while also helping the financial sector and technology incumbents to understand the latest developments in FinTech.

Bibliography

[1] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[2] M. A. Chen, Q. Wu, and B. Yang, "How valuable is fintech innovation?" *The Review of Financial Studies*, vol. 32, no. 5, pp. 2062–2106, 2019.

[3] L. Xu, X. Lu, G. Yang, and B. Shi, "Identifying fintech innovations with patent data: A combination of textual analysis and machine-learning techniques," in *Int. Conference on Information*. Springer, 2020, pp. 835–843.

[4] D. W. Arner, J. Barberis, and R. P. Buckey, "Fintech, regtech, and the reconceptualization of financial regulation," *Nw. J. Int'l L. & Bus.*, vol. 37, p. 371, 2016.

[5] D. Wójcik and T. F. Cojoianu, "A global overview from a geographical perspective," *Int. Financ. Centres after Glob. Financ. Cris. Brexit*, vol. 207, 2018.

[6] T. F. Cojoianu, G. L. Clark, A. G. Hoepner, V. Pazitka, and D. Wojcik, "Fin vs. tech: Determinants of fintech start-up emergence and innovation in the financial services incumbent sector," *Tech: Determinants of Fintech Start-Up Emergence and Innovation in the Financial Services Incumbent Sector*, 2019.

[7] P. Gomber, J.-A. Koch, and M. Siering, "Digital finance and fintech: current research and future research directions," *Journal of Business Economics*, vol. 87, no. 5, pp. 537–580, 2017.

[8] I. Goldstein, W. Jiang, and G. A. Karolyi, "To fintech and beyond," *The Review of Financial Studies*, vol. 32, no. 5, pp. 1647–1661, 2019.

[9] P. Gomber, R. J. Kauffman, C. Parker, and B. W. Weber, "Financial information systems and the fintech revolution," 2018.

[10] C. J. Fall, A. Törcsvári, K. Benzineb, and G. Karetka, "Automated categorization in the international patent classification," in *Acm Sigir Forum*, vol. 37, no. 1. ACM New York, NY, USA, 2003, pp. 10–25.

[11] K. Benzineb and J. Guyot, "Automated patent classification," in *Current challenges in patent information retrieval*. Springer, 2011, pp. 239–261.

[12] J. C. Gomez and M.-F. Moens, "A survey of automated hierarchical classification of patents," in *Prof. search in the modern world*. Springer, 2014, pp. 215–249.

[13] M. F. Grawe, C. A. Martins, and A. G. Bonfante, "Automated patent classification using word embedding," in *16th IEEE Int. Conference on Machine Learning and Applications (ICMLA)*. IEEE, 2017, pp. 408–411.

[14] S. Li, J. Hu, Y. Cui, and J. Hu, "Deeppatent: patent classification with convolutional neural networks and word embedding," *Scientometrics*, vol. 117, no. 2, pp. 721–744, 2018.

[15] M. Shalaby, J. Stutzki, M. Schubert, and S. Günnemann, "An lstm approach to patent classification based on fixed hierarchy vectors," in *Proceedings of the 2018 SIAM Int. Conference on Data Mining*. SIAM, 2018, pp. 495–503.

[16] J. Hu, S. Li, J. Hu, and G. Yang, "A hierarchical feature extraction model for multi-label mechanical patent classification," *Sustainability*, vol. 10, no. 1, p. 219, 2018.

[17] L. Abdelgawad, P. Kluegl, E. Genc, S. Falkner, and F. Hutter, "Optimizing neural networks for patent classification," in *Proc. of ECML-PKDD*, vol. 16, 2019.

[18] J.-S. Lee and J. Hsiang, "Patentbert: Patent classification with fine-tuning a pre-trained bert model," *arXiv preprint arXiv:1906.02124*, 2019.

[19] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural Comp.*, vol. 9, no. 8, 1997.

[20] T. Mikolov, K. Chen, G. Corrado, and J. Dean, "Efficient estimation of word representations in vector space," *arXiv preprint arXiv:1301.3781*, 2013.

[21] J. Risch and R. Krestel, "Domain-specific word embeddings for patent classification," *Data Technologies and Applications*, 2019.

[22] P. Bojanowski, E. Grave, A. Joulin, and T. Mikolov, "Enriching word vectors with subword information," *Trans. of the ACL*, vol. 5, pp. 135–146, 2017.

[23] J. Pennington, R. Socher, and C. Manning, "Glove: Global vectors for word representation," in *EMNLP*, vol. 14, 2014, pp. 1532–1543.

[24] Y. Kim, "Convolutional neural networks for sentence classification," *arXiv preprint arXiv:1408.5882*, 2014.

[25] S. Chae and J. Gim, "A study on trend analysis of applicants based on patent classification systems," *Information*, vol. 10, p. 364, 2019.

[26] M. Sofean, H. Aras, and A. Alrifai, "Analyzing trending technological areas of patents," in *Int. Conf. on DEXA*. Springer, 2019, pp. 141–146.

[27] K. Markus and B. Marcus, "Systems and methods for general aggregation of characteristics and key figures," U.S. Patent No 20 060 069 632, 2006.

[28] S. V. Bacastow and E. F. Arnold III, "Method and system for secure mobile remittance," U.S. Patent No 20 130 282 567, 2013.

[29] W. E. Bond Jr., M. A. Jackowski, H. Hezarkhani, J. J. Allogio, C. S. Benes, M. R. Krans, and J. Doyle, "Automated assignment of insurable events," U.S. Patent No 20 040 225 535, 2004.

[30] R. V. Subbu, K. Chalermkraivuth, J. R. Celaya, J. G. Russo, J. A. Ellis, H.-H. Doan, M. Ialeggio, and M. Allen, "Multi-criteria decision support tool interface, methods and apparatus," U.S. Patent No 20 080 163 085, 2008.

[31] J. Tumminaro, C. Realini, P. Hosokawa, D. Schwartz, S. Shawki, and N. V. Shah, "Mobile person-to-person payment system," U.S. Patent No 20 070 255 653, 2007.

[32] P. Maenpaa, J. Kessler, J. Moore, E. Kieran, L. Chiola, and J. Dallesandro, "System and method for generating and storing digital receipts for electronic shopping," U.S. Patent No 20 140 244 462, 2014.

[33] B. Mellon, "Innovation in payments: The future is fintech," *The Bank of Newyork*, 2015.

[34] R. Levy, "Fintech market map," *Retrieved*, vol. 11, no. 08, p. 2016, 2015.

[35] F. Amalia, "The fintech book: The financial technology handbook for investors, entrepreneurs and visionaries," *Journal of Indonesian Economy and Business*, vol. 31, no. 3, pp. 345–348, 2016.

[36] E. . Young and G. B. Treasury, *UK Fintech on the Cutting Edge: An Evaluation of the International Fintech Sector*. Ernst & Young.

[37] CBInsights, "Wealth tech market map," 2017.

[38] J. Eckenrode and S. Friedman, "Fintech by the numbers," *Deloitte Services LP.*, 2017.

[39] C. Haddad and L. Hornuf, "The emergence of the global fintech market: Economic and technological determinants," *Small Business Economics*, vol. 53, no. 1, pp. 81–105, 2019.

[40] C. Sun, X. Qiu, Y. Xu, and X. Huang, "How to fine-tune bert for text classification?" in *Nat. Conf. on Chinese Comp. Linguistics*. Springer, 2019, pp. 194–206.

[41] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[42] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized bert pretraining approach," *arXiv preprint arXiv:1907.11692*, 2019.

[43] Z. Lan, M. Chen, S. Goodman, K. Gimpel, P. Sharma, and R. Soricut, "Albert: A lite bert for self-supervised learning of language representations," in *Int. Conference on Learning Representations (ICLR)*, 2020.

[44] Z. Yang, Z. Dai, Y. Yang, J. Carbonell, R. R. Salakhutdinov, and Q. V. Le, "Xlnet: Generalized autoregressive pretraining for language understanding," in *Advances in NIPS*, 2019, pp. 5754–5764.

[45] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," 2019.

[46] I. Beltagy, M. E. Peters, and A. Cohan, "Longformer: The long-document transformer," 2020.

[47] P. He, X. Liu, J. Gao, and W. Chen, "Deberta: Decoding-enhanced bert with disentangled attention," 2020.

[48] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, and I. Sutskever, "Language models are unsupervised multitask learners," *OpenAI Blog*, vol. 1, no. 8, p. 9, 2019.

[49] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, and J. Brew, "Huggingface's transformers: State-of-the-art natural language processing," *ArXiv*, vol. abs/1910.03771, 2019.